WEST

Freeform Search

Database: US Patents Full-Text Database
US Pre-Grant Publication Full-Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term:

Display: 10 **Documents in Display Format:** TI **Starting with Number** 1

Generate: ○ **Hit List** ◉ **Hit Count** ○ **Side by Side** ○ **Image**

Search | Clear | Help | Logout | Interrupt

Main Menu | Show S Numbers | Edit S Numbers | Preferences | Cases

Search History

DATE: Friday, December 13, 2002 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT; PLUR=YES; OP=ADJ			
L18	l17 and (flush$4 or cache or modif$4 or updat$4 or tag or buffer or queue)	22	L18
L17	(3848234| 4713751| 5043870| 5045996| 5056002| 5136700| 5197146| 5214770| 5253353| 5261066| 5276836| 5276848| 5301287| 5357623| 5434993| 5524233| 5581727| 5603005| 5644753| 5666514| 5737757| 5778433)![pn]	22	L17
L16	L14 and l13	54	L16
L15	L14 and l13 and l12	5	L15
L14	(("L1" or upper or higher or first) adj2 cache) near8 (flush$4 or swap$4 or push$4) near8 (("L2" or lower or second) adj2 cache)	63	L14
L13	cache near6 (modif$8 or updat$4 or dirty or (store adj in))	5284	L13
L12	(flush or swap) adj2 buffer	598	L12
L11	L10 and ((711/122 |711/133 |711/135)!.CCLS.)	35	L11
L10	L9 and l3	213	L10
L9	buffer near4 (flush$3 or clear$3 or clean$3 or empt$4 or remov$4 or eliminat$4) near8 cache	330	L9
L8	l5 and (buffer near4 (flush$3 or clear$3 or clean$3 or empt$4 or remov$4 or eliminat$4))	600	L8
L7	L6 and l1	1	L7
L6	L5 and (buffer near4 flush$4)	184	L6
L5	L4 and l3	2637	L5
L4	cache near8 (flush$3 or clear$3 or clean$3 or empt$4 or remov$4 or eliminat$4)	4541	L4
L3	cache near8 (modif$8 or updat$4 or (store adj in))	5299	L3
L2	L1 and (cache near8 (modif$8 or updat$4 or (store adj in)))	8	L2
L1	(4349871 or 4442487 or 4525777 or 4755930 or 4794521 or 4843542 or 4860192 or 5023776 or 5025365 or 5205366).pn.	10	L1

END OF SEARCH HISTORY

Search Results

Search Results for: [(store in cache) and ((buffer or queue or register) <near/8> cache) and (tag <near/8> (stor* or updat* or modif*))]
Found 12 of 104,171 searched. → Rerun within the Portal

Search within Results

GO > Advanced Search > Search Help/Tips

Sort by: **Title** **Publication** **Publication Date** **Score** Binder

Results 1 - 12 of 12 **short listing**

1 Classifying load and store instructions for memory renaming 96%
Glenn Reinman , Brad Calder , Dean Tullsen , Gary Tyson , Todd Austin
Proceedings of the 13th international conference on Supercomputing May 1999

2 Memory dependence prediction using store sets 84%
George Z. Chrysos , Joel S. Emer
ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture April 1998 Volume 26 Issue 3
For maximum performance, an out-of-order processor must issue load instructions as early as possible, while avoiding memory-order violations with prior store instructions that write to the same memory location. One approach is to use memory dependence prediction to identify the stores upon which a load depends, and communicate that information to the instruction scheduler. We designate the set of stores upon which each load has depended as the load's "store set". The processor can discover and u ...

3 Memory forwarding: enabling aggressive layout optimizations by guaranteeing the safety of data relocation 80%
Chi-Keung Luk , Todd C. Mowry
ACM SIGARCH Computer Architecture News , Proceedings of the 26th annual international symposium on Computer architecture May 1999

Volume 27 Issue 2

By optimizing data layout at run-time, we can potentially enhance the performance of caches by actively creating spatial locality, facilitating prefetching, and avoiding cache conflicts and false sharing. Unfortunately, it is extremely difficult to guarantee that such optimizations are *safe* in practice on today's machines, since accurately updating *all* pointers to an object requires perfect alias information, which is well beyond the scope of the compiler for languages such as C. T ...

4 Implementation of precise interrupts in pipelined processors 80%

James E. Smith , Andrew R. Pleszkun

25 years of the international symposia on Computer architecture (selected papers) August 1998

5 Cache replacement with dynamic exclusion 80%

Scott McFarling

ACM SIGARCH Computer Architecture News , Proceedings of the 19th annual international symposium on Computer architecture April 1992 Volume 20 Issue 2

Most recent cache designs use direct-mapped caches to provide the fast access time required by modern high speed CPU's. Unfortunately, direct-mapped caches have higher miss rates than set-associative caches, largely because direct-mapped caches are more sensitive to conflicts between items needed frequently in the same phase of program execution. This paper presents a new technique for reducing direct-mapped cache misses caused by conflicts for a particular cache line. A small fi ...

6 Instruction fetch energy reduction using loop caches for embedded applications with small tight loops 77%

Lea Hwang Lee , Bill Moyer , John Arends

Proceedings 1999 international symposium on Low power electronics and design August 1999

7 Tolerating late memory traps in ILP processors 77%

Xiaogang Qiu , Michel Dubois

ACM SIGARCH Computer Architecture News , Proceedings of the 26th annual international symposium on Computer architecture May 1999 Volume 27 Issue 2

ILP processors can execute a large number of instructions at the same time. Thus it becomes more and more difficult to support traps efficiently. On the other hand a current trend in architecture is to support various memory functions in software rather than hardware, usually by trapping the execution processor on a cache

miss, TLB miss or a failed access to a local or remote memory. These late memory traps block the faulting instruction at the top of the active list, backing up the pipeline. Mo ...

8 A novel renaming scheme to exploit value temporal locality through physical register reuse and unification 77%
Stephen Jourdan , Ronny Ronen , Michael Bekerman , Bishara Shomar , Adi Yoaz
Proceedings of the 31st annual ACM/IEEE international symposium on Microarchitecture November 1998

9 Active memory: a new abstraction for memory system simulation 77%
Alvin R. Lebeck , David A. Wood
ACM Transactions on Modeling and Computer Simulation (TOMACS) January 1997
Volume 7 Issue 1

10 Don't use the page number, but a pointer to it 77%
André Seznec
ACM SIGARCH Computer Architecture News , Proceedings of the 23rd annual international symposium on Computer architecture May 1996
Volume 24 Issue 2

Most newly announced high performance microprocessors support 64-bit virtual addresses and the width of physical addresses is also growing. As a result, the size of the address tags in the L1 cache is increasing. The impact of on chip area is particularly dramatic when small block sizes are used. At the same time, the performance of high performance microprocessors depends more and more on the accuracy of branch prediction and for reasons similar to those in the case of caches the size of the Br ...

11 Recovery protocols for shared memory database systems 77%
Lory D. Molesky , Krithi Ramamritham
ACM SIGMOD Record , Proceedings of the 1995 ACM SIGMOD international conference on Management of data May 1995
Volume 36 Issue 7

12 On reconfigurable on-chip data caches 77%
Fredrik Dahlgren , Per Stenström
Proceedings of the 24th annual international symposium on Microarchitecture September 1991

Results 1 - 12 of 12 **short listing**